AR-009-473

DSTO-GD-0076

Software Implementation of a Timecode Reader for Modified Irig B Format

Adam D. Sbrana

APPROVED FOR PUBLIC RELEASE

DEPARTMENT OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

Software Implementation of a Timecode Reader for Modified IRIG B Format

Adam D. Sbrana

Maritime Operations Division
Aeronautical and Maritime Research Laboratory

DSTO-GD-0076

ABSTRACT

Timecode readers form an essential part of many data analysis and processing schemes, and are generally designed to read an analogue timing code from a ferromagnetic tape. At the present time most data is digitised before it is processed further, and so a method of reading the original analogue timing code after it has been digitised is necessary in order to establish an accurate time standard between events which can only be examined in different digitised datasets. This document describes the software implementation of a timecode reader with improved accuracy and flexibility over conventional analogue timecode readers for temporally localising events in a digitised dataset.

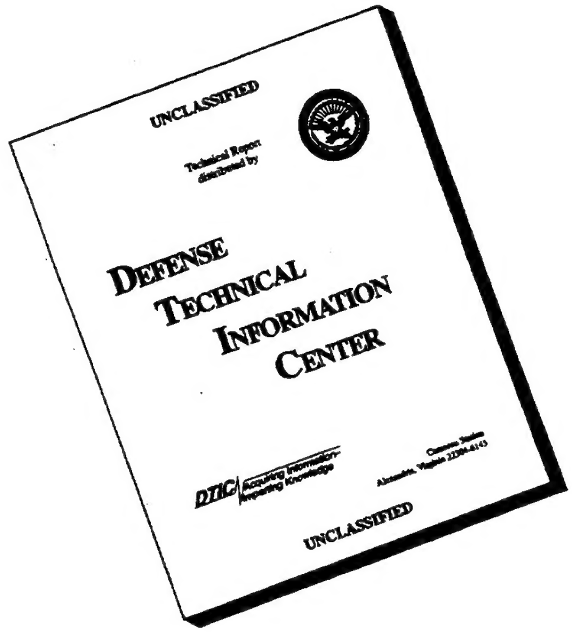
RELEASE LIMITATION

Approved for public release.

DEPARTMENT OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Published by

DSTO Aeronautical and Maritime Research Laboratory
PO 4331
Melbourne.Victoria, 3001, Australia

Telephone: *(08) 259 5209*
Fax: *(08) 259 7115*

AR-009-473
February, 1996

APPROVED FOR PUBLIC RELEASE

Software Implementation of a Timecode Reader for Modified IRIG B Format

EXECUTIVE SUMMARY

The software described in this paper was developed specifically to assist in the localisation, segmentation and correlation of passive sonar data by accurately determining event times. The timecode reader is not limited to these tasks as it can be used in conjunction with any data set which has been digitised in parallel with an IRIG B timecode to accurately time the events recorded in the data. The software was written for unix. However it has been designed to be portable. The software has also been written using object orientated methodologies which make it well suited to be used in conjuction with other software. The software is driven by command line inputs from the user and provides two levels of help for the novice and the expert.

Contents

1. Introduction

The deterioration of magnetic tape by stretching, coupled with non contiguous data segments, means that it is not a simple matter to interpolate a Zulu time for discrete points taken along a magnetic tape. If the timecode from a typical reel of recorded analogue data is examined in detail, there will normally be many corruptions to the timecode track. These corruptions manifest themselves through an analogue timecode reader as periods in which the exact time is unknown.

This document was written as a result of problems discovered during work carried out concerning signal classification and localisation tasks. When detecting and segmenting signals from large amounts of data, it is common to mark the events with reference points in the file. This is adequate for crude data segmentation but when accurate time delays between signals are required it becomes necessary to estimate the Zulu time of to reload the magnetic tape and re-examine the data. Re-examination of the tape is a poor solution and monitoring the timecode second by second is a waste of time, especially as there is very likely to be many corruptions to the timecode. Another set back with this approach is that an analogue tape reader provides an accuracy of approximately half a second which is sufficient for rough estimation (or cataloguing) but inadequate for fine segmentation or localisation purposes.

A solution to these two problems can be obtained by writing software to read the digitised timecode. This represents little to no overhead (provided the storage medium is fast enough both in terms of hardware and accessibility) since the majority of data is now analysed using digital computers. This naturally means that the timecode channel must be digitised in parallel with as many data tracks as is practicable and is necessary in order to preserve the association. The advantages of using a software version of a timecode reader are many. Firstly the accuracy of a reading depends purely upon the data, which for an IRIG B format means 1mS (see Appendix B). It is a simple matter to search for a particular Zulu time or offset in seconds from the start of the file. An analogue timecode reader does not perform searches for particular offsets from a set marker, it only reads Zulu time directly from the time track. Due to the nature of the medium it is reading, it must access it sequentially. The software timecode reader on the other hand does not have to cycle through vast amounts of tape to read a particular time hence its throughput is much greater. This increase in throughput and greater accuracy will also speed the process of localisation, where it is imperative to have high resolution Zulu times of events from many tracks and to find them quickly.

2. Software structure

The software can be logically partitioned into two sections, the first section controlling the second. The first section is responsible for recursively searching for a desired Zulu time or controlling errors which may occur, usually due to corrupted timecode packets. The second section is responsible for reading the Zulu time at a given point specified by the first section.

2.1 Modified IRIG B data format

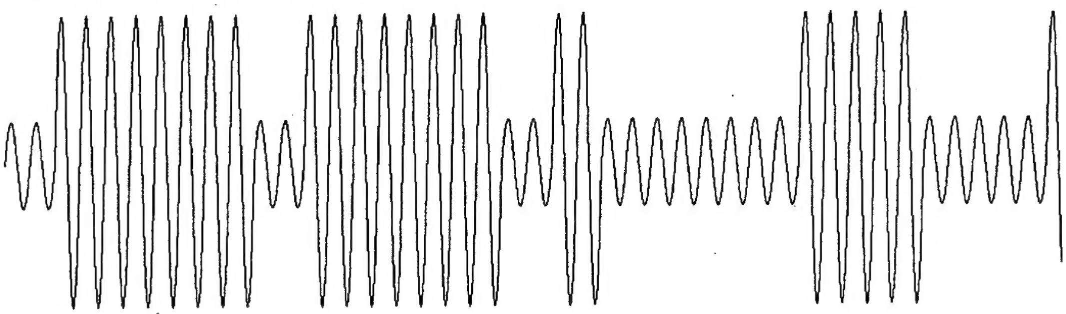


Figure 1: Theoretical timecode segment

The format of IRIG B timecode is outlined in appendix B and shown in figure 1. The modulation of the data is PWM (Pulse Width Modulation), the carrier frequency is 1kHz. Typical digitisation rates, chosen for their Nyquist frequency include 8kHz and 16kHz. At these frequencies there are between 8 and 16 samples per cycle. For the data on which this code was modelled the sampling rate was 8kHz and a segment of timecode at this sampling rate is shown in figure 2.

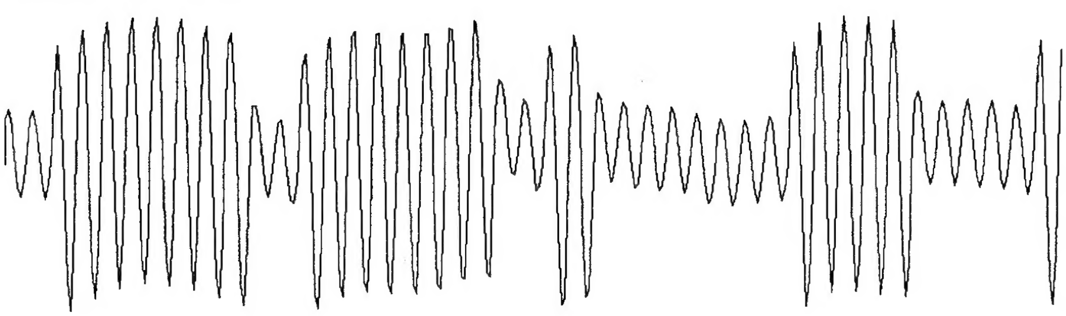


Figure 2: Realistic timecode segment

The first notable difference between the theoretical and realistic data segments is the data envelope, which is shown in figure 3. This follows an exponential growth and decay caused by the filter characteristics of the digitising hardware. The second notable difference is the somewhat jagged appearance of the carrier wave, although this is hard to see due to the image capturing technique. Both of these characteristics can readily be overcome as will be shown.

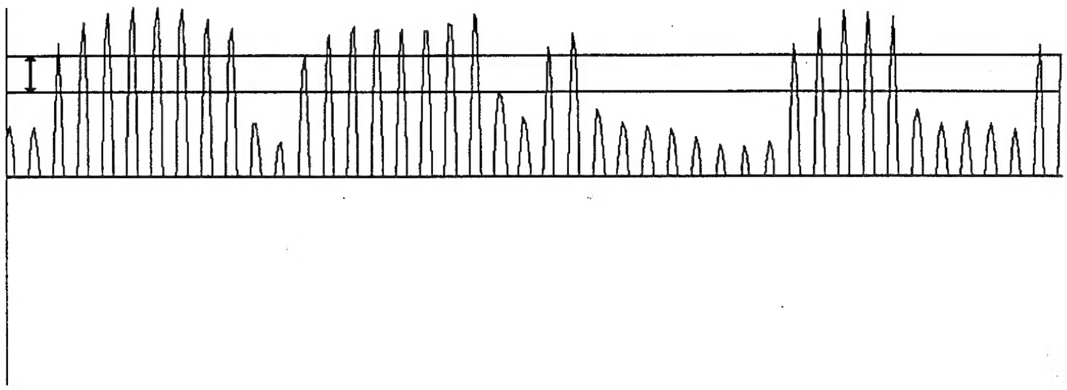


Figure 3: Timecode envelope

2.2 Search methods

2.2.1 Searching by an offset

The way in which this section of code operates depends upon the mode of the search. If the search mode is 'by seconds', indicating the user wants to know a Zulu time at a particular offset from the start of the timecode file, then this offset is simply passed to the section of code that does a 'once off' Zulu time read for Zulu time determination. If an error occurs then recursive searches will be performed either side of the requested offset at predefined, or user set intervals until the timecode file is exhausted, or a valid time can be read. An estimate of the Zulu time at the requested offset will then be given based upon the difference between the requested and uncorrupted read times.

This search is shown graphically in figure 4.

2.2.2 Searching by a Zulu time

The alternative search mode is 'by Zulu'. This mode is more involved due to the necessity of recursive searches. The first stage is to determine a reference time: unless the user specifies otherwise, the middle of the file is selected. The Zulu time at this point is compared with the passed Zulu time and an updated search time determined. This continues until the Zulu time is found to the accuracy desired; as selected by the

user, or the timecode file is exhausted. If errors occur reading the timecode file then recursive searches are performed identical to a search 'by seconds'. Should the end of the file be reached before the successful termination criterion have been met then one of two things will happen. If a valid time was read anywhere during the analysis then the last valid time will be used to present an estimated offset; if no valid time was read then no offset can be presented.

Figure 5 graphically depicts searching for a Zulu time.

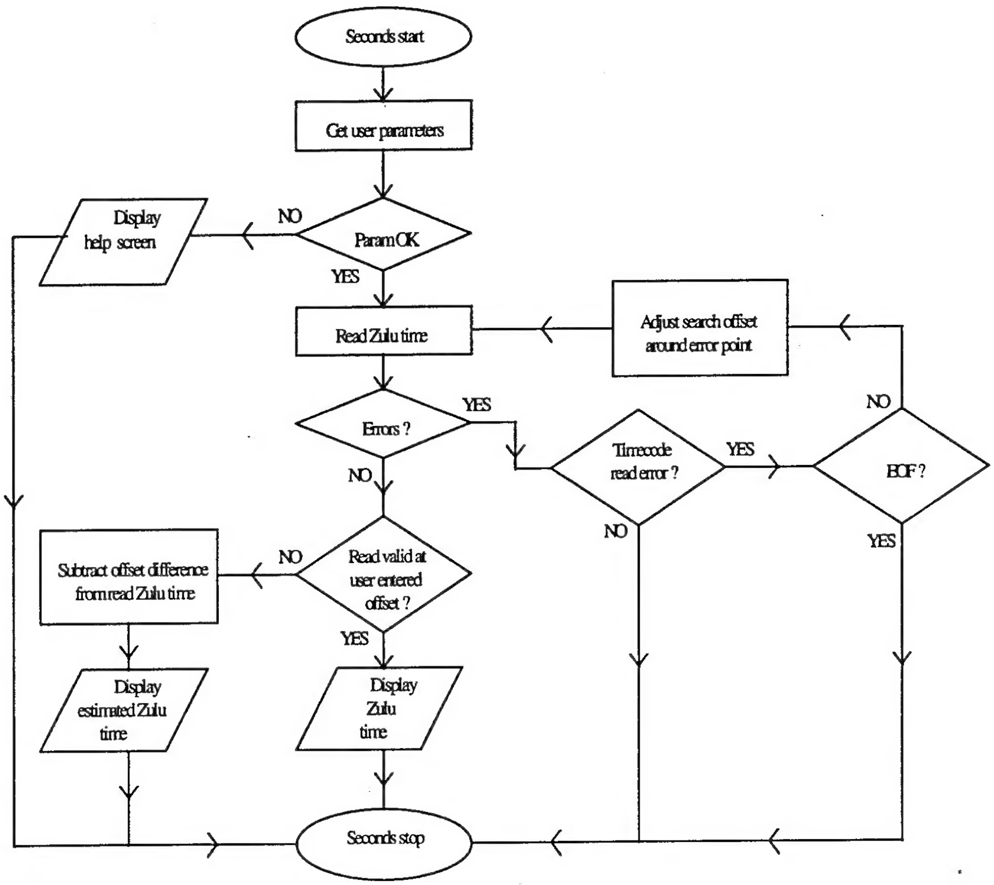


Figure 4: Search for an offset in seconds

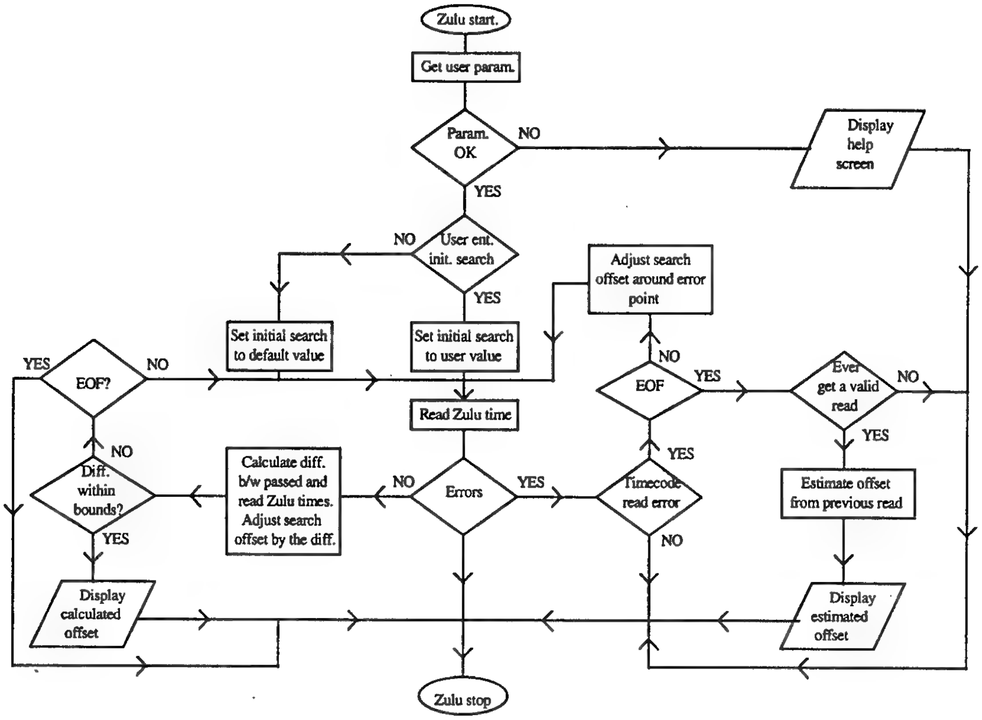


Figure 5: *Search for a Zulu time*

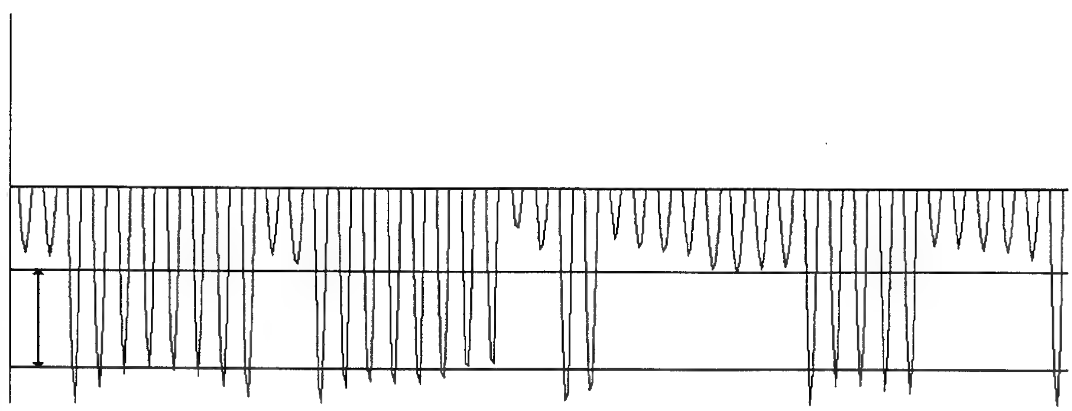


Figure 6: *Lower timecode envelope*

2.3 Single time read algorithm

The Zulu time reading software works on a simple algorithm as shown in figure 7. Initially, sufficient data is read to cover two complete cycles of IRIG B data. This is done so that it will be of no concern where the data packet commences, at least one complete packet will always be present. The next step is to demodulate the signal. This is accomplished by simply picking all of the peaks in the read data. The data must them be converted to a binary format before being decoded. This is achieved with the aid of a software Schmitt trigger.

Referring to figure 2 and figure 3 it is evident that there is a great deal of variability in the waveform at points that should be classified as 0 and 1 respectively and there is often minor variability between crossover points. This makes even a Schmitt trigger difficult to implement without generating errors. However if we look at the lower envelope (figure 6), we can see that the shape is much sharper, especially in the transition regions, producing a much greater safety margin[1]. Thus picking the negative peaks presents a more viable solution.

Once the demodulation has been completed for all of the data samples the Zulu time is ready to be extracted. The first step in the extraction is to find the reference marker (Appendix B). Unfortunately this pulse width has the same duration as the position identifying markers and, as such, requires a recursive search for two consecutive pulses of the same duration; namely the position identifier zero and the reference marker. Once the reference marker has been located the five sub-packets of interest (from P0 to P5), which give seconds, minutes, hours and days, can be read. Decoding the five sub-packets simply requires sliding a 10mS window across the binary data. Being PWM the pulse width determines whether the magnitude at any particular position is assigned a one or a zero weighting. Provided the timecode is not corrupted the Zulu time is then known. The time at the start of the raw data read during the first step can be calculated from knowledge about the carrier frequency and the number of sample to the reference pulse. The final step, which must be selected by the user, involves correcting the search offset and adjustment parameters, utilising the actual IRIG B carrier frequency decoded from the data stream by peak picking. This step assumes that the IRIG B carrier frequency on the timecode generator was precise, and the all irregularities that happen to the timecode channel from the point of analogue recording to analysing via work-stations happen in exactly the same fashion to every data channel.

[1] The safety margin is the area bounded by the two horizontal lines and arrow.

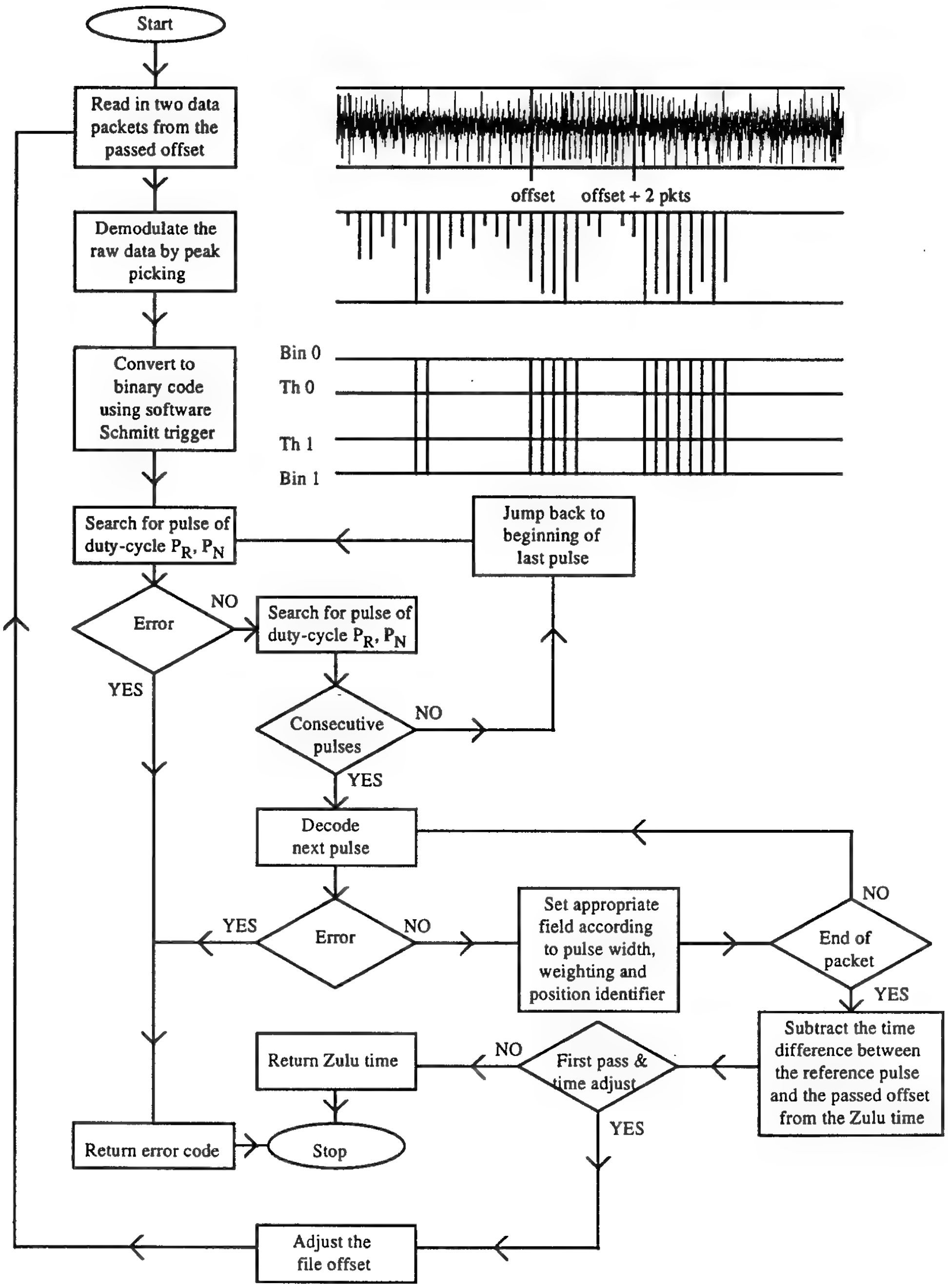


Figure 7: Single Zulu time reader flowchart

3. Technical considerations

In standard mode the accuracy of the returned Zulu time is conditional upon the accuracy and precision of the digitisation rate. Search offsets are calculated from the file sampling rate assuming it is accurate and stable. As briefly outlined in section 2.3 there is a software switch that will disregard the assumed sampling rate and force the algorithm to calculate the utilised sampling rate. The adjustment is performed in order to match the extracted carrier frequency to the ideal IRIG B carrier frequency of 1kHz. This is accomplished by counting the peaks over a preset time frame and calculating the detected IRIG B carrier frequency by peak picking. The two frequencies are then compared and adjustments made to the sampling rate used. The search is then performed once more and a more precise reading taken.

This technique will not produce reliable results on data that contains localised discrepancies, since the time frame over which the digitised IRIG B carrier frequency is calculated is only a small section of the data stream.

Since the time adjustment option forces one more pass through the Zulu reading algorithm it does slow the process of determining a time down by approximately a factor of two. However if this option is not used, calculation will show that for every 1% variation in sampling rate (assuming 8kHz nominal) the Zulu time returned will be in error 36.36 seconds for every hour of data.

4. Discussion

Due to the modularity of this software it will be a simple case to incorporate modules to decode other timecode formats should this be deemed necessary at a later date. The construction of the software also lends itself to easy incorporation with other code such as automatic segmenting and localisation algorithms which are currently being written

At any location on the earth, at the same instant of time the Zulu time (Greenwich Mean Time) is identical, thus providing a consistent reference. The transfer and up-keep of this time on different platforms proves to be inconsistent and there can be a difference of several seconds between different platforms, ie. a submarine and aircraft. While this may seem trivial it does result in time being wasted when correlating data from these different platforms, especially for tasks such as range estimation.

AMADEUS[2] can be used to determine the closest point of arrival (CPA) of a target to a sensor. AMADEUS can also provide estimates of the sound pressure level (SPL) of a target. By using all available evidence, including estimates of CPA and other parameters provided by AMADEUS, it is sometimes possible to determine the time differences between sensors on platforms even if the timecode is corrupted. Once an incident can be located temporally, a new timecode can be written to go with the dataset containing the incident.

Inaccuracies in the speed of the analogue tape deck and digitisation equipment are not trivial concerns. With the aid of this software package we were able to calculate that certain transcripts of our end data[3] were 0.3% in error. While this figure is quite low it does represent a reasonably large error in range estimation. This problem was overcome is software by comparing the calculated and true IRIG B carrier frequencies and making adjustments accordingly.

With the aid of the timecode reader an estimation of the sampling rate at the point of analysis can be determined. This result can then be used in any analysis rather than the nominal sampling rate of the data. Thus errors involved in producing the digitised data can be reduced.

The mechanisms of laying down timecode tracks upon a storage medium are not infallible, hence there will often be segments where the timecode is unreadable. These corrupted segments can be from milliseconds to several seconds. Should a time be

---

[2] AMADEUS (Accurate Measurement And Description of Underwater Sound) is a software package developed specifically for the analysis of underwater sound detected on passive sonars (single channel or beamformed aural data). The package was designed by Dr. Darryl McMahon and coded by Grant Schwarz of MOD, DSTO Salisbury, SA.

[3] After recording on the relevant platform, replay and digitisation in the laboratory

desired within one of these corrupted segments, time estimation can be used as an alternative. This software will provide such estimation provided that it can read an uncorrupted time in the vicinity. The accuracy of this method is subject to the two main problems indicated in the introduction, that is contiguous data and tape stretching. Another method would be to overlay a timecode upon a corrupted timecode. With minor modifications this software could achieve this, but again this method is subject to tape conditions.

5. Conclusions

This paper has presented a viable, expeditious alternative to analogue tape readers with far improved accuracy and flexibility. The timecode reader has proven very successful in accurately cataloguing vast numbers of signals with ease.

Since most data storage systems have or are moving toward digital techniques such as CD's, digital timecode readers will become more prevalent and prove quite useful either as stand alone units or in conjunction with analogue readers.

THIS PAGE INTENTIONALLY BLANK

Appendix A

A1 Commands and switches, simple usage

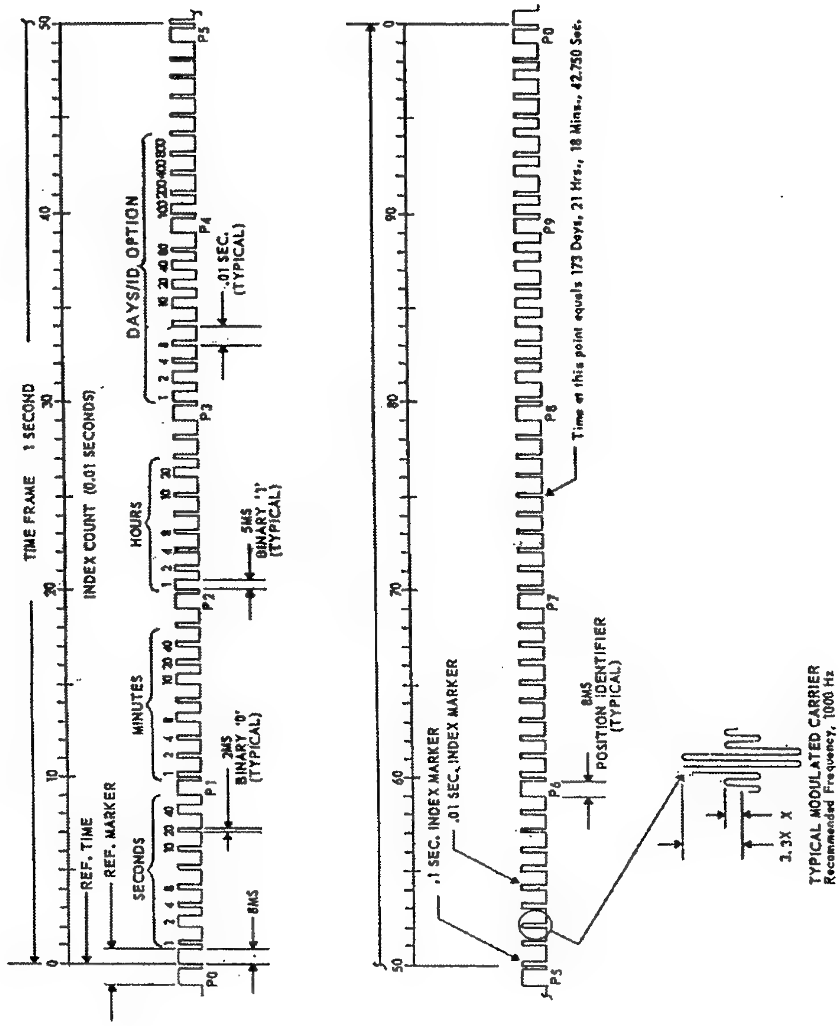
```
MOD AMRL DSTO Salisbury S.A. 5th September 1995.

Timecode reader V8.0 supporting the following formats:
-> IRIG B

Usage
timecode <command> <data> <filename> <-switches>
Eg. timecode irigb z 173,21:49:30.342 ah169tk10tc.8000
Eg. timecode irigb s 332.45 ah169tk10tc.8000


<Commands>
h : Full help.
z : Zulu time follows, determine how many seconds into the file
    this Zulu time is.
s : Seconds into file follows, determine the Zulu time at this point.

<Data format>
Zulu time: days,hrs:min:sec.frac sec
other    : unit.fractional
```

A2 Commands and switches, complete usage

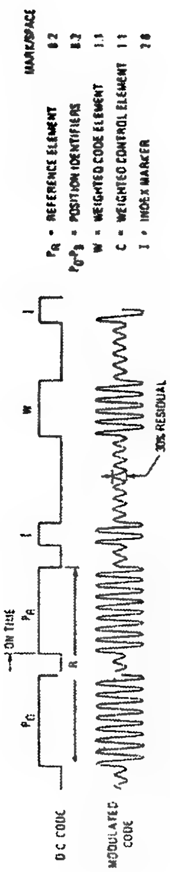
```
MOD AMRL DSTO Salisbury S.A. 5th September 1995.

Timecode reader V8.0 supporting the following formats:
-> IRIG B

Usage
timecode <command> <data> <filename> <-switches>
Eg. timecode irigb z 173,21:49:30.342 ah169tk10tc.8000 -df -tl 0.597
    -tu 0.85 -sr 16000 -md


<Commands>
z : Zulu time follows, determine how many seconds into the file
    this Zulu time is.
s : Seconds into file follows, determine the Zulu time at this point.

<Data format>
Zulu time: days,hrs:min:sec.frac sec
other    : unit.fractional

<Switches>
ta: This adjusts the returned Zulu time or offset as necessary
    according to the difference between the digitised IRIG B carrier
    frequency and the nominal IRIG B carrier frequency.  When performing
    a search for a Zulu time this may cause a number of oscillations due
    to slightly different carrier frequencies being read at each iteration.
ds: Demodulation using single peak detection and getting other peaks
    from knowledge about the sampling rate and carrier frequency.
    This method should only be used if both frequencies are accurate
    and precise.
sr: Digitised sampling rate, taken from file extension if not supplied.
tl: Lower threshold multiplier (of max peak) for binary Schmitt
    trigger conversion.  Default is 0.45000.
tu: Upper threshold multiplier (of max peak) for binary Schmitt
    trigger conversion.  Default is 0.68000.
ts: Greatest mismatch in seconds allowed when performing a Zulu search.
    Default value is 0.00100 seconds
os: Incremental offset in seconds for a search when an error is
    encountered at the desired search point.
    For a search by Zulu it is 20.00000 seconds
    and for a search by seconds it is  1.00000 seconds
is: The initial search conducted during a search by Zulu is the file mid-point
    This value can be explicity set using this switch
md: Don't match days when performing Zulu search.
mh: Don't match hours when performing Zulu search.
mm: Don't match minutes when performing Zulu search.
ms: Don't match seconds when performing Zulu search.
df: Full diagnostic, show every step. (Recommended redirection to file).
dd: Demodulation diagnostic. (Recommended redirection to file).
db: Binary conversion diagnostic. (Recommended redirection to file).
dt: Time read diagnostic. (Recommended redirection to file).
da: Time adjustment diagnostic.
```

Appendix B

B1 Modified IRIG B packet format



B2 Modified IRIG B weighting ratios and specifications



R = REFERENCE MARKER THE ON TIME POINT TO WHICH THE CODE REFERS IS THE LEADING EDGE OF P_R

CODE	TIME FRAME	CODE LENGTH	ELEMENT RATE	TIME AT A-A (INDEX COUNT 31)				CARRIER FREQUENCY
A	0.1 sec	100 bits	1,000 p.p.s	173 days	21 hours	18 min	42.831 sec	10 kHz
B	1 sec	100 bits	100 p.p.s	173 days	21 hours	18 min	42.31 sec	1 kHz
C	1 min	120 bits	2 p.p.s	173 days	21 hours	18 min	15½ sec	1 kHz
D	1 hour	60 bits	1 p.p.m	173 days	21 hours	32 min		1 kHz
E	10 sec	100 bits	10 p.p.s	173 days	21 hours	18 min	43.1 sec	1 kHz
G	0.01 sec	100 bits	10,000 p.p.s.	173 days	21 hours	18 min	42.8031 sec	100 kHz
H	1 min	60 bits	1 p.p.s	173 days	21 hours	18 min	31 sec	1 kHz

SOFTWARE IMPLEMENTATION OF A TIMECODE READER FOR MODIFIED IRIG B FORMAT (U)

Adam D. Sbrana

DSTO-GD-0076

DISTRIBUTION LIST

	Number of Copies
DEPARTMENT OF DEFENCE	
Defence Science and Technology Organisation	
Chief Defence Scientist and members of the)	1 shared copy
DSTO Central Office Executive)	for circulation
Counsellor, Defence Science, London	(Document Control sheet)
Counsellor, Defence Science, Washington	(Document Control sheet)
Scientific Adviser - POLCOM)	1 shared copy
Senior Defence Scientific Adviser)	
Director, Aeronautical & Maritime Research Laboratory	1
Maritime Operations Division	
Chief, Maritime Operations Division	1
Research Leader, Sonar Technology and Processing	1
Research Leader, Mine Warfare Branch	1
Research Leader, Operational Environment	1
Research Leader, Maritime Operations Research	1
Head, Advanced Sonar Processing Techniques	1
Head, Active Sonar	1
Head, Airborne Processing and Sonar Systems	1
Head, Combat Systems Research Centre	1
Head, Sonar Technology	1
Head, Trials and Data Analysis	1
Head, Sonar Systems	1
Head, Sonar Environment Group	1
Head, Mine Warfare Operations	1
Head, Mine Warfare Signatures	1
Head, Mine Warfare Sonars	1
Head, Sonar Environment	1
Head, Above Water Warfare	1
Head, Under Water Warfare	1

Head, Torpedo Countermeasures and Simulation	1
Head, Submarine Operations	1
Author (A. Sbrana)	1
D. McMahon	1
S. Godfrey	1
R. MacLeod	1
K. Amos	1
M. Archer	1
D. Brown	1
I. Cox	1
G. Day	1
T. Puust	1
T. Jackson	1
P. Jackson	1
C. Webster	1
M. Englund	1
G. Brown	1
A. Grigorakis	1
C. Lightowler	1
C. Martin	1
J. Marwood	1
E. Parker	1
L.Kelly	1
G. Furnell	1
P. Roberts	1
T. Trainor	1
S. Taylor	1
D. Walker	1
C. Gillard	1
A. Tynan	1
P. Farr	1
M. Steed	1
I. Woon	1
S. Hoefs	1
G. Speechley	1
C. Davis	1

Ship Structures and Materials Division

Chief, Ship Structures and Materials Division	1

Sponsor

ASSTASS-PD (CP3-4-04)	1

Libraries and Information Services

Defence Central Library - Technical Reports Centre	1
Manager Document Exchange Centre (MDEC) (for retention)	1
Additional copies which are to be sent through MDEC	
Defence Science and Technology Organisation Salisbury, Research Library	2
Defence Science and Technology Organisation Pyrmont, Research Library	2
Defence Science and Technology Organisation Maribynong, Research Library	2
Defence Intelligence Organisation	1
Library Defence Signals Directorate Canberra	1
AGPS	1

Navy Office

Navy Scientific Adviser (A-1-16)	1

Airforce Office

Airforce Scientific Adviser (C-4-41)	1

Maritime Headquaters

Scientific Officer-Maritime	1
Manager, Range and Assessing Unit (MHQ-Annex)	1
CO, Australian Joint Acoustic Analysis Centre (HMAS Albatross)	1

Spares	5

1. Page Classification UNCLASSIFIED
2. Privacy Marking/Caveat (of document)

Department of Defence
DOCUMENT CONTROL DATA SHEET

3a. AR Number AR-009-473	3b. Laboratory Number DSTO-GD-0076	3c. Type of Report GENERAL DOCUMENT	4. Task Number M40507
5. Document Date DECEMBER 1995	6. Cost Code 370	7. Security Classification	8. No of Pages 16
10. Title Software Implementation of a Timecode Reader for Modified IRIG B Format		* U U U Document Title Abstract S (Secret) C (Confi) R (Rest) U (Unclas * For UNCLASSIFIED docs with a secondary distribution LIMITATION, use (L) in document box.	9. No of Refs 0
11. Author(s) Adam D. Sbrana		12. Downgrading/Delimiting Instructions	
13a. Corporate Author and Address Aeronautical and Maritime Research Laboratory PO Box 4331,Melbourne, Vic 3001		14. Officer/Position responsible for Security:..............CMOD.............. Downgrading: ....CMOD.............. Approval for Release: ..CMOD..............	
13b. Task Sponsor ASSTASS PD			

15. Secondary Release Statement of this Document

Approved for public release

16a. Deliberate Announcement

No limitation

16b. Casual Announcement (for citation in other documents)

[X] No Limitation [ ] Ref. by Author, Doc No. and date only

17. DEFTEST Descriptors	18. DISCAT Subject Codes
Timecode readers; Data analysis IRIG B	

19. Abstract
Timecode readers form an essential part of many data analysis and processing schemes, and are generally designed to read an analogue timing code from a ferromagnetic tape. At the present time most data is digitised before it is processed further, and so a method of reading the original analogue timing code after it has been digitised is necessary in order to establish an accurate time standard between events which can only be examined in different digitised datasets. This document describes the software implementation of a timecode reader with improved accuracy and flexibility over conventional analogue timecode readers for temporally localising events in a digitised dataset.